Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2005 Proceedings

Americas Conference on Information Systems (AMCIS)

2005

The Spread of Internet Worms and the Optimal Patch Release Strategies

Peng Han
University of Washington, penghan@u.washington.edu

Yong Tan
University of Washington, YTAN@U.WASHINGTON.EDU

The Spread of Internet Worms and the Optimal Patch Release Strategies

Peng Han
University of Washington Business School,
Box 353200, Seattle, Washington 98195-3200
penghan@u.washington.edu

Yong Tan
University of Washington Business School,
Box 353200, Seattle, Washington 98195-3200
ytan@u.washington.edu

ABSTRACT

Internet worms spread in an automated fashion and can cause tremendous damage in a short period. As worms start spreading, knowing the worm prevalence patterns under the impact of various patching schemes is important for software vendors to decide whether or not, and when to release the patches. Due to the strong analogy between the spread of worms on Internet and the spread of disease among human society, we analytically model the spreading process and the impact of patching decisions on it by using the same techniques in epidemiological research. We find that, only by releasing patches providing immunity to susceptible users, the epidemic can be ceased efficiently. From the viewpoint of software vendors, the patch development cost and the reputation cost incurred indirectly from victim users should be balanced to decide whether, and when, the patch should be released. The paper gives closed form solutions for the optimal patch release time and discusses the conditions in which the patch should not be released. The results in this paper can be used either as a starting point for further research, or by software vendors for deciding their patch release strategies.

Keywords

Security strategy, Internet worm, Epidemiological model, Patch.

INTRODUCTION

As Internet becomes more and more pervasive, worms pose an ever-bigger threat to the computer society. Different from viruses, which usually attach themselves to other files or programs, worms exist as separate entities. They spread themselves actively over the network from one computer to the next by taking advantage of automatic file sharing and network services, which makes them difficult to be identified and removed. In 2003, for instants, MSBlast exploited a flaw with the Remote Procedure Call (RPC) process, and quickly spread out over Internet. Variants of the worm caused computer networks around the world to collapse. A particularly vicious version of the worm unleashed in August 2003 infected more than 48,000 computers running Microsoft Windows systems in a short period (Shukovsky 2005). In addition to the tremendous loss among end users, it also caused a huge reputation cost on the system vendor side — a 768 million dollars drop in Microsoft's sales to large corporation (INQUIRER 2003), which is partly attributed to the outbreak of the worm. Due to the active spreading nature, the defense strategies against worms are not quite the same as against computer virus. In addition to release corrective patches that remove worms and fix infected systems, software vendors usually offer preventive patches to repair the system flaw and provide immunity to those who have not yet been attacked. For example, Microsoft periodically offers system patches to Windows system user for protecting them from being exploited by certain worms. However, considering the fact that there are more than twelve thousand worms out on the Internet (Sophos 2005), it is important to know the worm prevalence patterns under the impact of various patching schemes before software vendors deciding whether or not, and when to release the patches. Furthermore, the expected damage caused by certain worm and the reputation cost incurred indirectly from victim users should be evaluated against the development cost in order to find the optimal patch release time such that the vendor's total cost is minimized.

Despite the difference in definition and existence status, Internet worms and computer viruses share many similarities in terms of random spreading characteristics. Due to the strong analogy between the propagation of computer virus and biological virus, epidemiology models are widely used in modeling the prevalence of virus and worms in network environment (Cohen 1987, Murray 1988). Kephart and White (1991, 1993) studied the spread of computer virus with a Susceptible-Infected-Susceptible (SIS) model, where a computer can be infected and cured repeatedly, while Wang (2000), Zou (2002), and Kim (2004) further enriched the virus propagation literature with a Susceptible-Infected-Removed (SIR) model, which assumes a node cannot be infected again once the virus or worm is removed. Both models are widely adopted and perform well in predicting the virus or worm prevalence in reality. For instants, SIS model fits the scenarios that, the

victims manually remove the worm without effectively applying preventive patches. Therefore the system is still susceptible to the same worm. However, in case that appropriate patches are used, the users will not only get rid of the worm, but also be immune from getting infected again, therefore SIR model applies.

To the best of our knowledge, however, little of previous research addresses the situation that susceptible nodes actively download and apply preventive patches before the worms even hit them. This is especially true when the worms spread aggressively and the potential damage is huge. For instants, as MSBlast worm blasted across the web in 2003, alert was raised via Internet, newspapers, and TV news to urge all computer users download and install the patches from Microsoft's web site (BBC News 2003). As a result, the preventive patches prevent both infectious and susceptible from getting infected in future. Obviously the virus spreading speed is directly linked with the time by which the patches are released and the rate at which users adopt the preventive patches.

Another issue particularly interesting to software vendors is that, given that a worm's spreading pattern and potential damage are discovered, whether or not worth to develop a patch for it; and if yes, how soon the patch should be released. Faster development and release can reduce or cease the epidemic earlier, but will cost more on the vendor side in development. On the other hand, later patch release eases the development but leaves larger window for the worm to spread out. Although, in many cases, the damage on user side does not impose direct cost to vendors, it usually incurs certain reputation cost, which may undercut the vendors' revenue in future. This timing issue of patch release has not been extensively studied. Arora et al. (2004) study the optimal policy for software vulnerability disclosure under a game-theoretic framework. Cavusoglu et al. (2004) analyze the tradeoff between patch development duration and potential damage caused by hackers or virus. Their studies do not particularly consider the worm spreading process over Internet, and the curing process is assumed to be instantly done after the patch is released. However, due to the random spreading feature, it usually takes a relatively long time to cease the epidemic even when the patch is available.

In the rest of this paper, we model the spreading process of worms under different circumstance in following section, where four types of patches, namely, no patch, corrective patch, preventive patch, and comprehensive patch, and their impact on the worm propagation process, are studied. We show that comprehensive patch has obvious superiority over others in terms of maximum infection level and duration of epidemic. In section 3, we particularly address the optimal patch release problem for comprehensively patching scheme, and derive closed form solutions such that the vendor's total cost is minimized. We also discuss the conditions under which the patch is immediately released, and the patch is never released. The impact of different parameters on the optimal patch release time is studied at the same time. Finally we conclude the paper in section 4 and discuss future directions.

THE WORM SPREADING MODEL

Here we assume only one kind of worm is under consideration. Following the same setup as in Kephart (1991), the worm spreads in a random graph that has totally N nodes, which are assumed to be vulnerable to the worm. The edges represent the links between computers along which the worm spreads. An infectious node, a node that has been exploited by the worm, infects its vulnerable neighbors with a rate of β. In following subsections, we first revisit the classic SIS model, also the simplest case, where no official patch is available. Then we move on to the cases where various types of patches, namely, corrective patch, preventive patch, and comprehensive patch, are available. The worm spreading processes under different cases are compared and discussed.

Worm Spreads without Patching

Without loss of generality, we assume that, although no patch can be applied, some of the victim users still can manually remove the worm by themselves. Assume the average cure rate for each infected node is δ_m. We also assume that, once an individual is cured, it is immediately capable of being re-exploited by the same worm. Denote $I(t)$ as the number of infectious nodes at any time t, and $i(t) = I(t)/N$, the fraction of infectious nodes. The time evolution of $i(t)$ can be captured by following classic SIS model:

$$\frac{di}{dt} = \beta i(1-i) - \delta_m i\,, \tag{1}$$

The solution to differential equation (1) is:

$$i(t) = \frac{i_0(1-\rho)}{i_0 + (1-\rho-i_0)e^{-(\beta-\delta_m)t}}\,, \tag{2}$$

where $\rho = \delta_m / \beta$, is the average ratio of the rate at which an infected node is cured to that at which it infects other nodes, and $i_0 = i(t = 0)$ is the initial fraction of infected nodes.

Then we can easily derive the limiting values of i as follows:

$$\lim_{t\to\infty} i(t) = \begin{cases} 0, & \text{as } \rho \geq 1, \\ 1-\rho, & \text{as } \rho < 1. \end{cases} \tag{3}$$

It says that, when the cure rate is higher than the infection rate, i decays exponentially from i_0 to 0. However, if the cure rate is lower than the infection rate, i goes from i_0 and eventually saturates to $(1 - \rho)$ which is between 0 and 1. The intuition behind this is straightforward. If the number of neighbors that an infectious node can exploit is less than one, the epidemic will die out by itself. If this number is greater than one, the worm will spread out, but not necessarily infect all population at the end. SIS model successfully captures the diffusion dynamics in case that there is no official treatment available as Internet worms spread, though some victims may be able to remove the worms by themselves.

Worm Spreads with Patching

As software vendors increasingly realize the seriousness of Internet worms, it is often the case that patches addressing particular worms will be released soon after it starts spreading. By applying the patches, users can either fix the damage caused by the worms, or be immune from being infected by the same worms in future. The patches usually fall into three categories, namely corrective patch, preventive patch, and comprehensive patch. A corrective patch is a removal tool specifically targeting certain worms. It efficiently cleans up the worms from infected system but, usually, does not protect users from being re-exploited. Differently, preventive patch usually cannot fix the infected systems but can provide immunity for those who have not yet infected by the worms. For instants, as MSBlast Worm spread in 2003, Windows users could download and install the preventive patch from Microsoft web site to protect them from being infected (ZDNet 2003). However the infected users could not get rid of the worm by applying the patch. Comprehensive patch is basically a combination of above two kinds of patches. By applying a single comprehensive patch, infected users can get their systems fixed, while susceptible users can get immunized.

Corrective Patch

Since the system cured by applying corrective patches will be susceptible for re-infection immediately, the spreading model here is essentially the same as the SIS model shown previously, except the average cure rate is replaced by δ_{cr}, the average rate at which the corrective patch is applied. The subscript *cr* means corrective. We can safely assume that $\delta_{cr} > \delta_m$ because removing the worm with corrective patch should be much easier than doing it manually. We also assume that when corrective patch is available, no infected user would choose to manually remove the worm by himself/herself. The system dynamics then can be described by following differential equation:

$$\frac{di_{cr}}{dt} = \beta i_{cr}(1 - i_{cr}) - \delta_{cr} i_{cr} \,. \tag{4}$$

Similarly, we can solve above differential equation and get

$$i_{cr}(t) = \frac{i_0(1-\rho_{cr})}{i_0 + (1 - \rho_{cr} - i_0)e^{-(\beta-\delta_{cr})t}},$$

where $\rho_{cr} = \delta_{cr} / \beta$.

It is easy to see that, by applying corrective patch, the spreading process is essentially the same as no patch being applied, except the limiting prevalence level is lower (given $\delta_{cr} > \delta_m$). Therefore, solely applying the corrective patch will not cease the propagation and the epidemic will persist.

Preventive Patch

When preventive patches are available, the susceptible users can get themselves protected before being infected by the worm. Therefore the susceptible population decreases as more and more patches are adopted. Denote the number (fraction) of the susceptible nodes by $S_p(t)$ ($s_p(t)$). The subscript *p* means preventive. And assume the average rate at which susceptible users adopt the preventive patch is δ_p. The systems dynamics can be described by following coupled differential equations:

$$\begin{cases} \dfrac{di_p}{dt} = \beta i_p s_p - \delta_m i_p, \\ \dfrac{ds_p}{dt} = -\beta i_p s_p + \delta_m i_p - \delta_p s_p. \end{cases} \tag{5}$$

Note in this model, we assume that the preventive patch does not help infected users remove the worm. Therefore the cure rate is δ_m here.

Above coupled differential equations do not have closed form solutions. However, by observing the dynamics of s_p, we can see that, due to the preventive patch, the susceptible population keeps decreasing along the spreading process. Eventually, the whole population will become immune and the epidemic will stop.

Comprehensive Patch

It is often the case that the software vendor provides both corrective patch and preventive patch together, or packs them into one single patch, namely comprehensive patch, which not only removes the worm from infected system, but also provides immunity for susceptible users. In such scenario, the recovered users will not be susceptible again after applying the comprehensive patch. A comprehensive patch is equivalent to a corrective patch for infected users, and a preventive patch for susceptible users. The average rates at which the patch is adopted by the two kinds of users can be different. However, considering the fact that the patching process is increasing automated, here we assume both kinds of users will adopt the patch at the same rate, δ_c. The subscript c means comprehensive. Then the system dynamics can be captured as follows:

$$\begin{cases} \dfrac{di_c}{dt} = \beta i_c s_c - \delta_c i_c, \\ \dfrac{ds_c}{dt} = -(\beta i_c + \delta_c) s_c. \end{cases} \tag{6}$$

i_c can be solved from above differential equations as follows,

$$i_c(t) = \frac{i_0 e^{-\delta_c t}}{i_0 + (1 - i_0) e^{\frac{\beta}{\delta_c}\left(e^{-\delta_c t} - 1\right)}}. \tag{7}$$

Since comprehensive patch combines the functionalities of both corrective patch and preventive patch, it is expected to perform better in reducing the maximum spreading level as well as ceasing the epidemic. Figure 1 compares the spreading process under different cases, where N = 100, β = 1.0, δ_m = 0.05, δ_{cr} = 0.2, δ_p = 0.1, δ_c = 0.1, and i_0 = 0.01. The parameter values that used in the numerical study are hypothetical but reasonable. Similar parameter configuration has also been used in Kephart and White (1991), and Kim et al. (2004).

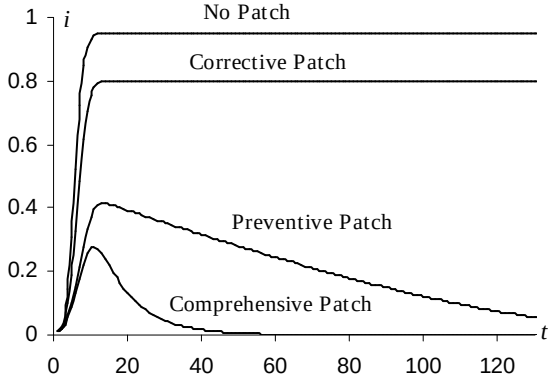

Figure 1. Comparison of virus spreading process under different conditions

Figure 1 shows that the worm spreads out and the epidemic converges to certain level if the patch does not provide immunity and the cure rate is less than the infection rate. However, once immunity can be provided, the epidemic will be reduced greatly and the peak of prevalence is limited by a much lower value. The spreading process diminishes after the peak. In

addition, comprehensive patch shows strong superiority over other patching schemes in terms of maximum prevalence level and the duration of the epidemic.

OPTIMAL PATCHING STRATEGIES

Due to the obvious advantage of comprehensively patching, we focus only on the case where comprehensive patch are applied from now on. Specifically, we look for the optimal patch release time such that the total vendor cost is minimized for given period [0, T]. Timely release of patches requires intense development effort, and hence incurs high development cost. However, earlier patch release helps prevent the outbreak of the worms and reduces the damage among users, which may affect the vendor's long-term reputation and future revenues. The optimal strategy should balance these effects to minimize the total cost.

Cost Structure

On the user side, the cost is directly linked with the degradation of the system performance caused by the worm. In previous literatures, such cost is usually measured by a one-time cost per infection without considering the duration of infection (Cavusoglu et al. 2004). However, in real world, the length of time during which the system is in imperfect condition should be taken into account. Here we denote γ as the cost incurred by infection during each time unit. Within a time unit, therefore, the whole system incurs a cost of $\gamma i(t)$ at any time t. And the total user cost, C_u, is

$$C_u = \gamma \int_0^T i(t)dt \ . \tag{8}$$

On the vendor side, there are two categories of costs incurred. The cost for developing the comprehensive patch, and the reputation cost indirectly transferred from the damage on the user side, i.e. the user cost.

Following Cavusoglu et al. (2004) and Arora et al. (2004), we model the patch development cost $C_d = c - \tau\varepsilon$, where c is the development cost if the patch is released right after the worm starts spreading, and $\tau\varepsilon$ represents the savings in patch development cost associated with delaying the release to time τ. As in Cavusoglu et al. (2004), we assume ε is small enough compared with c so that the total development cost will not be negative. This cost definition captures the basic relationship between the intensity of patch development and the cost associated with it.

The reputation cost is closely related to the cost incurred on the user side. The more performance degradation caused by the worm, the higher the reputation cost will be. Therefore we denote the reputation cost as αC_u, where $\alpha > 0$ (Cavusoglu et al. 2004).

Therefore, the total vendor cost, C, is

$$C(\tau) = c - \tau\varepsilon + \alpha C_u \ . \tag{9}$$

Assuming the patch is released at time τ, the spreading process before and after τ is captured by equation (1) and (6), respectively. Denote the user cost in each period as C_1 and C_2, respectively. Then we have

$$C_u = C_1 + C_2 = \gamma \int_0^\tau i(t)dt + \gamma \int_\tau^T i_c(t,\tau)dt \ .$$

Note that the spreading process after the patch is released, i_c, is a function of both t and τ.

Since the epidemic will diminish as comprehensive patch is applied, the total user cost must be bounded. Therefore, without introducing discount effect, we can extend T to infinity and still have the problem solvable. Rewrite the total vendor cost as

$$C(\tau) = c - \tau\varepsilon + \alpha\gamma\left(\int_0^\tau i(t)dt + \int_\tau^\infty i_c(t,\tau)dt \right). \tag{10}$$

Resolving (6) with initial conditions $i_c(\tau) = i(\tau)$, and $i_c(\tau) + s_c(\tau) = 1$, we have

$$i_c(t,\tau) = \frac{i(\tau)e^{\delta_c(\tau - t)}}{i(\tau) + (1 - i(\tau))e^{\frac{\beta}{\delta_c}\left(e^{\delta_c(\tau - t)} - 1\right)}} \ . \tag{11}$$

With (2) and (11) we can solve C_1 and C_2 as

$$C_1 = \frac{\gamma}{\beta}\ln\left(1 + \frac{\beta i_0 (e^{(\beta-\delta_m)\tau} - 1)}{\beta - \delta_m}\right), \text{ and } C_2 = \frac{\gamma}{\beta}\ln\left(1 + (e^{\beta/\delta_c} - 1)i(\tau)\right). \quad (12)$$

Optimal Release Time

Formally, we define the vendor cost minimization problem as

$$\min_{\tau} C(\tau) = c - \tau\varepsilon + \alpha\gamma\left(\int_0^{\tau} i(t)dt + \int_{\tau}^{\infty} i_c(t,\tau)dt\right), \quad (13)$$

By FOC, we have

$$C'(\tau) = -\varepsilon + \alpha\gamma\left(i(\tau) + \frac{(e^{\beta/\delta_c} - 1)\left(\beta i(\tau) - \beta i^2(\tau) - \delta_m i(\tau)\right)}{\beta(1 + (e^{\beta/\delta_c} - 1)i(\tau))}\right) = 0, \quad (14)$$

which is a linear function about $i(\tau)$. Solving it, we get $i^*(\tau)$ as

$$i^*(\tau) = \kappa \equiv \frac{\varepsilon}{\alpha\gamma}\left(1 + (e^{\beta/\delta_c} - 1)\left(1 - \rho - \frac{\varepsilon}{\alpha\gamma}\right)\right)^{-1}, \quad (15)$$

where $\rho = \delta_m / \beta$. From (15) and (2), we can solve the optimal patch release time as

$$\tau^* = \frac{1}{\beta - \delta_m}\ln\frac{\kappa(1 - \rho - i_0)}{i_0(1 - \rho - \kappa)}. \quad (16)$$

Figure 2 shows the change of vendor's costs as τ increases, where N = 100,000, i_0 = 0.00001, β = 5.0, δ_m = 0.5, δ_c = 3.0, α = 0.5, γ = 10, c = 20, and ε = 2.

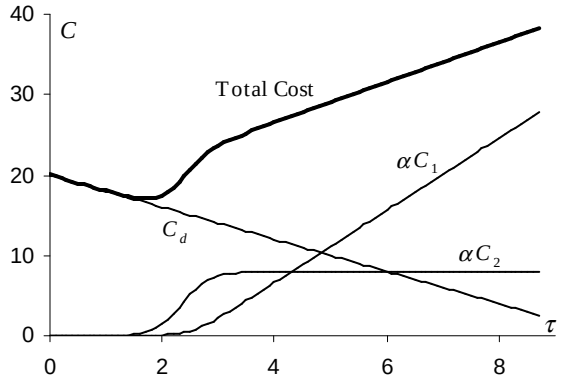

Figure 2. Vendor's Cost versus the patch release time

However, it is not always the case that the vendor will release the patch after the worm spreads for some time. Intuitively, if the savings from delaying the release is large enough, the vendor will choose not to release the patch. On the other hand, if the savings is far less than the reputation cost, the vendor may release the patch immediately after the worm starts spreading. In order to find out the boundary conditions, we need to study the feasible range of κ. From (3) we know that $i(\tau)$ is bounded by i_0 and $1 - \rho$ (here only consider the case that $\rho < 1$). The same boundaries should hold for κ, because κ is the prevalence level just before the patch is released. Therefore, $\kappa \le i_0$ is the condition under which the vendor will release the patch immediately, while $\kappa \ge 1 - \rho$ is the condition under which the vendor will never release the patch.

From the perspective of the vendor, it would be more straightforward to see these conditions expressed by the relative significance of the unit delay savings on development cost, ε, to the reputation cost incurred by each infection, $\alpha\gamma$. Therefore, we solve the threshold values of $\varepsilon/(\alpha\gamma)$ from above conditions, and formally present the optimal solutions as

$$\tau^* = \begin{cases} 0, & \dfrac{\varepsilon}{\alpha\gamma} \le B_l; \\ \dfrac{1}{\beta - \delta_m} \ln \dfrac{\kappa(1-\rho-i_0)}{i_0(1-\rho-\kappa)}, & B_l < \dfrac{\varepsilon}{\alpha\gamma} < B_u; \\ \infty, & \dfrac{\varepsilon}{\alpha\gamma} \ge B_u. \end{cases} \tag{17}$$

where

$$B_l = \frac{1 + (e^{\beta/\delta_c} - 1)(1-\rho)}{1/i_0 + (e^{\beta/\delta_c} - 1)};$$

$$B_u = \frac{1 + (e^{\beta/\delta_c} - 1)(1-\rho)}{1/(1-\rho) + (e^{\beta/\delta_c} - 1)}.$$

Since $i_0 \le 1 - \rho$, $B_l \le B_u$ always holds. Above solutions give a clear guideline for the vendor to make patching decisions. In case that $B_l = B_u$, there is no optimal release time problem. The vendor will either release the patch instantly, or never release any patch. It is easy to prove that, when $\varepsilon/(\alpha\gamma) < B_l$, κ must be greater then zero. When $\varepsilon/(\alpha\gamma) > B_u$, however, κ could be either greater than $1 - \rho$, or less than zero. Therefore the previous boundary conditions on κ should be revised to

$$\tau^* = \begin{cases} 0, & 0 < \kappa \le i_0; \\ \dfrac{1}{\beta - \delta_m} \ln \dfrac{\kappa(1-\rho-i_0)}{i_0(1-\rho-\kappa)}, & i_0 < \kappa < 1-\rho; \\ \infty, & \kappa \ge 1-\rho, \text{ or } \kappa \le 0. \end{cases} \tag{18}$$

Based on the same parameter configuration as in Figure 2, Figure 3 shows the impact of $\varepsilon/(\alpha\gamma)$ on the optimal patch release time, where B_l and B_u are 0.00005 and 0.9, respectively.

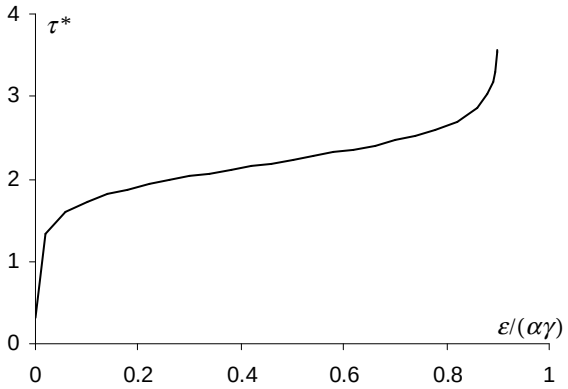

Figure 3. Impact of $\varepsilon/(\alpha\gamma)$ on the optimal patch release time

Figure 4 shows the impact of δ_m, the rate at which victim users manually remove the worm, on the optimal patch release time. As δ_m increases, the optimal time by which the patch should be released is convexly increasing. It says that when users can remove the worm by themselves more efficiently, the vendor will be less pressured to release a patch promptly. Therefore it can delay the patch release for a longer time to reduce the development cost. As δ_m exceeds certain threshold value, the vendor virtually does not need to release any patch at all, because the users are efficient enough in dealing with the worm by themselves. Figure 5 shows the impact of δ_c, the rate at which victim users remove the worm or get immunized by applying the comprehensive patch, on the optimal patch release time. τ^* is concavely increasing in δ_c, which means that, as the patch works more efficiently, the vendor is less likely to release the patch in a rush. However, no matter how large δ_c is, the vendor always needs to release the patch, given that δ_m is lower than the threshold value. If δ_c is below certain threshold value, the vendor will release the patch immediately. The underlying logic is that, since the patch is so inefficient in ceasing the epidemic, the vendor has to release it as soon as possible to avoid the worm spreading out. An interesting observation from these numerical results is that, the decision of whether or not *immediately* release the patch depends on δ_c, while the decision of whether or not release any patch at all depends on δ_m.

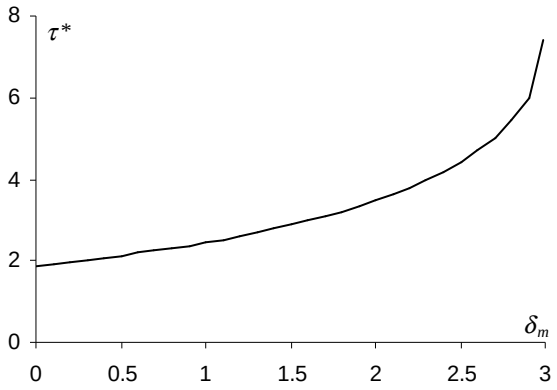

Figure 4. Impact of δ_m on the optimal patch release time

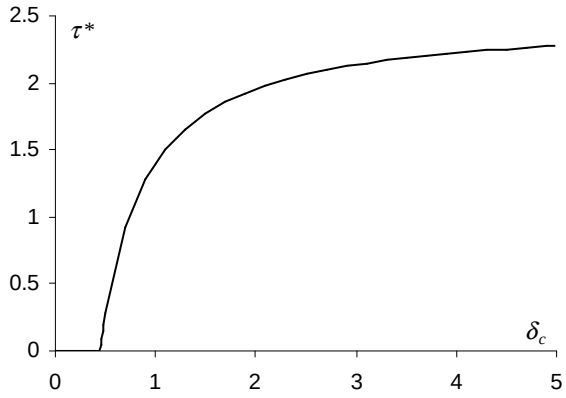

Figure 5. Impact of δ_c on the optimal patch release time

CONCLUSION AND FUTURE DIRECTIONS

In this paper, we study the spreading process of Internet worms and the impact of various patching schemes on it. Starting with the classic SIS model, we analyze the simplest case where no patch is available and infected users have to remove the worm manually. If the rate at which the worm is removed is less than the infection rate, the epidemic will persist and the limiting value of the prevalence level is $1 - \rho$, where ρ is the ratio of the average rate at which the worm is removed to the average infection rate. In case that there is corrective patch available, users can fix infected systems easier with the patch. Therefore the average cure rate is very likely to be higher and the limiting value of prevalence level will be lower than the case where no patch is available. However, since no immunity is provided, solely applying corrective patch does not cease the epidemic unless the cure rate exceeds the infection rate. When preventive patch, the patch that immunizes susceptible users, emerges, the number of vulnerable systems keeps decreasing as more and more users adopt the patch. The epidemic will finally die out even though preventive patch does not cure infected systems directly. Finally, as a combination of corrective patch and preventive patch, we introduce the notion of comprehensive patch, which cures and immunizes adopters at the same time. Numerical result shows that, by applying comprehensive patch, both maximum prevalence level and the duration of the epidemic can be reduced significantly.

Based on the spreading models developed, we further explore the optimal patch release problem in terms of vendor's total cost over infinite horizon. Due to the strong superiority of comprehensive patch, we limit our research to this patching scheme only. The optimal time by which the patch is released balances the patch development cost, which is assumed to be a linear decreasing function over time, and the reputation cost incurred by the users' damage caused by the worms. Different from previous IT security research, the user cost in this paper is modeled as a function of not only the number of infected users, but also the length of time during which the system is in imperfect condition. We find closed form solutions for the vendor cost minimization problem. The results suggest that the patch release time is convexly increasing in the rate at which the users manually remove the worms, and concavely increasing in the rate at which the comprehensive patch is adopted. We also derive the boundary conditions, under which the vendor will release the patch immediately after the prevalence starts, or never release the patch at all.

In addition to the findings in this research, there are plenty of room to improve and many interesting directions to go. For example, the development cost is simplified to a linear decreasing function of the duration of the development. Although it applies well when the duration is within certain range, it fails when τ is large enough so that the development cost becomes negative, which is clearly not true in reality. A more proper way to model the development cost should capture the feature of decreasing marginal benefit. Another refinement can be made is about the parameter values in the spreading models. At this time point, we assume that these values are predetermined and ready for use as the vendor makes patching decisions. In real world, however, it is not likely for the vendor to know about the parameters right after the worm starts spreading. Instead, it can only estimate them, and the accuracy of the estimation is closely related to the time the vendor spends in observing the propagation of the worm. In this case, delaying the patch release will not only reduce the development cost, but also help the vendor to get a clear picture of the spreading process. In this paper, we optimize the patch release time based on the total vendor cost. The user cost is considered as an indirect cost to the vendor and discounted by α. It could be interesting if we look at the problem from the viewpoint of social welfare. Certain incentive mechanisms might be needed to align the social optima and the vendor optima.

REFERENCES

1. Arora, A., R. Telang, and H. Xu (2004) "Optimal Policy for Software Vulnerability Disclosure." Working paper (Available online at http://www.dtc.umn.edu/weis2004/xu.pdf)
2. BBC News (2003) "Worm Blasts Across the Web." *BBC News*, August 12 (Available online at http://news.bbc.co.uk/1/hi/technology/3143625.stm).
3. Cavusoglu, H., H. Cavusoglu, and S. Raghunathan (2004) "How Should We Disclose Software Vulnerabilities?" *Proceedings of Fourteenth Annual Workshop on Information Technologies and Systems*, December 11-12, Washington, DC, pp. 243-248.
4. Chen, Z., L. Gao, and K. Kwiat (2003) "Modeling the Spread of Active Worms." *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, March, San Francisco, CA (Available online at http://citeseer.ist.psu.edu/chen03modeling.html).
5. Cohen, F. (1987) "Computer Viruses, Theory and Experiments." *Computers & Security*, Vol. 6, pp. 22-35.
6. INQUIRER (2003) "Blaster caused Microsoft to lose big sales." *The INQUIRER*, October 24 (Available online at http://www.theinquirer.net/?article=12313).
7. Kephart, J. O. (1994) "How Topology Affects Population Dynamics." *C. Langton, Ed., Artificial Life III. Studies in the Sciences of Complexity*, pp. 447-463.
8. Kephart, J. O. and S. R. White (1991) "Directed-Graph Epidemiological Models of Computer Viruses." *IEEE Computer Society Symposium on Research in Security and Privacy*, May 20-22, Oakland, California, pp. 343-359.
9. Kephart, J. O. and S. R. White (1993) "Measuring and Modeling Computer Virus Prevalence." *IEEE Computer Society Symposium on Research in Security and Privacy*, May 24-26, Oakland, California, pp. 2-15.
10. Kim, J., S. Radhakrishnan, and S. K. Dhall (2004) "Measurement and Analysis of Worm Propagation on Internet Network Topology." *Thirteenth International Conference on Computer Communications and Networks*, October 11-13, Chicago, IL, USA
11. Murray, W. H. (1988) "The Application of Epidemiology to Computer Viruses." *Computers & Security*, Vol. 7, pp. 130-150.
12. Sethi, S. P., and P. W. Staats (1978) "Optimal Control of Some Simple Deterministic Epidemic Models." *The Journal of the Operational Research Society*, Vol. 29, No. 2, pp. 129-136.
13. Shukovsky, P. (2005) "Blaster worm attacker gets 18 months." Seattle Post-intelligencer, January 29 (Available online at http://seattlepi.nwsource.com/local/209900_worm29.html).
14. Sophos (2004) "Sophos virus analyses,"

    Available online at http://www.sophos.com/virusinfo/analyses/index_st_worm.html
15. Wang, C., J. C. Knight, and M. C. Elder (2000) "On Computer Viral Infection and the Effect of Immunization." *Proceedings of the 16th Annual Computer Security Applications Conference*, pp. 246–256.
16. White, S. R. (1998) "Open Problems in Computer Virus Research." *Virus Bulletin Conference*, October, Munich, Germany, (Available online at

    http://www.research.ibm.com/antivirus/SciPapers/White/Problems/Problems.html).
17. ZDNet (2003) "Protecting yourself from the MSBlast worm." ZDNet, August 12 (Available online at http://insight.zdnet.co.uk/internet/security/0,39020457,39115633,00.htm)
18. Zou, C. C., W. Gong, D. Towsley (2002) "Code Red Worm Propagation Modeling and Analysis." *Ninth ACM Conference on Computer and Communication Security*, Nov. 18-22, Washington DC, USA.